

AD-A155 531 ANALYSIS TECHNIQUES FOR MICROWAVE DOSIMETRIC DATA(U)
TECHNOLOGY-USA INC OXON HILL MD M J CAMPBELL ET AL.
JUL 81 DAMD17-79-C-9151

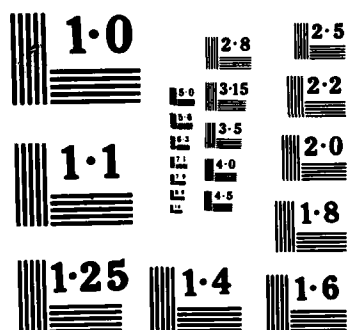
AD-A155 531 ANALYSIS TECHNIQUES FOR MICROWAVE DOSIMETRIC DATA(U)
TECHNOLOGY-USA INC OXON HILL MD M J CAMPBELL ET AL.
JUL 81 DAMD17-79-C-9151

1/1

UNCLASSIFIED

F/G 6/18

NL



NATIONAL BUREAU OF STANDARDS
MICROCOPY RESOLUTION TEST CHART

AD-A155 531

DTIC FILE COPY

AD _____

ANALYSIS TECHNIQUES FOR
MICROWAVE DOSIMETRIC DATA

ANNUAL REPORT

By
M.J. CAMPBELL, T.E. GOFF, and V.L. KALB

24 JULY 1981

Supported By
U.S. ARMY MEDICAL RESEARCH AND DEVELOPMENT COMMAND
Fort Detrick, Frederick, Maryland 21701

Contract No. DAMD17-79-C-9151

Technology-USA, Inc.
P.O. Box 55333 Oxon Hill, Maryland 20022

Approved for public release; distribution unlimited

The views, opinions, and/or findings contained in this report
are those of the authors and should not be construed as an
official Department of the Army position, policy, or decision,
unless so designated by other authorized documentation.

DTIC
JUN 20 1985
S D

G

85 5 22 059

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM										
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER										
4. TITLE (and Subtitle) RESEARCH IN ANALYSIS TECHNIQUES FOR MICROWAVE DOSIMETRIC DATA		5. TYPE OF REPORT & PERIOD COVERED Annual Report Oct 79 - June 81										
7. AUTHOR(s) M.J. Campbell, T.E. Goff and V.L. Kalb		6. PERFORMING ORG. REPORT NUMBER										
9. PERFORMING ORGANIZATION NAME AND ADDRESS Technology-USA Inc. Oxon Hill, MD 20022		8. CONTRACT OR GRANT NUMBER(s) DAMD17-79-C-9151										
11. CONTROLLING OFFICE NAME AND ADDRESS US Army Medical Research and Development Command Fort Detrick Frederick, MD 21701		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62771A.3E162771A805.00.011										
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE July 1981										
		13. NUMBER OF PAGES 43										
		15. SECURITY CLASS. (of this report) Unclassified										
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE										
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release, Distribution unlimited												
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		<table border="1"> <tr> <th colspan="2">Accession For</th> </tr> <tr> <td>NTIS GRA&I</td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>DTIC TAB</td> <td><input type="checkbox"/></td> </tr> <tr> <td>Unannounced</td> <td><input type="checkbox"/></td> </tr> <tr> <td>Justification</td> <td></td> </tr> </table>	Accession For		NTIS GRA&I	<input checked="" type="checkbox"/>	DTIC TAB	<input type="checkbox"/>	Unannounced	<input type="checkbox"/>	Justification	
Accession For												
NTIS GRA&I	<input checked="" type="checkbox"/>											
DTIC TAB	<input type="checkbox"/>											
Unannounced	<input type="checkbox"/>											
Justification												
18. SUPPLEMENTARY NOTES		<table border="1"> <tr> <td>By</td> <td></td> </tr> <tr> <td>Distribution/</td> <td></td> </tr> <tr> <td>Availability Codes</td> <td></td> </tr> </table>	By		Distribution/		Availability Codes					
By												
Distribution/												
Availability Codes												
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Image Processing Cubic Spline Interpolation Two Segment Enhancement Curve		<table border="1"> <tr> <th>Dist</th> <th>Avail and/or Special</th> </tr> <tr> <td>A/1</td> <td></td> </tr> </table>	Dist	Avail and/or Special	A/1							
Dist	Avail and/or Special											
A/1												
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The overall objective of this research investigation is to develop a methodology for displaying microwave dosimetric data. Special emphasis was placed on developing a method for evaluating the spatial variation of absorbed microwave energy by specific organs and organ subdivisions. Techniques were developed by a computer and displayed on an image in pseudocolor. Manipulation of the data using the image system allows the operator to highlight organ subdivisions within												

(Block 20 Continuation)

an image. A cubic spline interpolation program was developed to uniformly expand the data set for presentation on a large format display. A two segment enhancement curve was developed which gives the operator the capability of creating his own lookup table for pseudocolor translation of images stored in refresh memory. The results to date show that pseudocolor imaging is an effective method of analyzing microwave dosimetric data. Additional processing algorithms need to be developed and the system hardware should be optimized for image processing.

TABLE OF CONTENTS

<u>Section</u>		<u>Page</u>
I.	INTRODUCTION.....	6
II.	DEVELOPMENT OF DATA ANALYSIS SYSTEM.....	7
	A. Selection of a Display Media.....	7
	B. System Description.....	9
	C. Image System Operation.....	15
III.	IMAGE PROCESSING PROGRAMS.....	19
	A. Cubic Spline Interpolation.....	19
	B. Two Segment Enhancement Curve.....	21
IV.	APPENDIX.....	25
	A. Cubic Spline Interpolation Program Listing.....	25
	B. Cubic Spline Interpolation Program Documentation.....	33

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1.	Block Diagram of RM-9351 Image Display System.....	10
2.	RM-9351 Control/Video Section.....	12
3.	Example of Density Slicing.....	14
4.	Example of One Dimensional Interpolation.....	22
5.	Example of Two Dimensional Interpolation.....	23
6.	Two Segment Enhancement Curve.....	24

SUMMARY

The overall objective of this research investigation is to develop a methodology for displaying microwave dosimetric data. Special emphasis was placed on developing a method for evaluating the spatial variation of absorbed microwave energy by specific organs and organ subdivisions. Techniques were developed by a computer and displayed on an image in pseudocolor. Manipulation of the data using the image display system allows the operator to highlight organ subdivisions within an image. A cubic spline interpolation program was developed to uniformly expand the data set for presentation on a large format display. A two segment enhancement curve was developed which gives the operator the capability of creating his own lookup table for pseudocolor translation of images stored in refresh memory. The results to date show that pseudocolor imaging is an effective method of analyzing microwave dosimetric data. Additional processing algorithms need to be developed and the system hardware should be optimized for image processing.

SECTION I

Introduction

The research described in this Annual Report is the result of efforts performed during the first year of this contract. As a result of an inordinate delay in obtaining necessary hardware for this project, it was necessary to extend the time for performance of the contract beyond the original one year. The overall objectives of this research investigation is to develop a methodology for displaying microwave dosimetric data on a CRT type display for use in data interpretation and analysis. A color image processing system was procured and interfaced to the laboratory Hewlett-Packard computer system. Data obtained from the microwave scanner is processed by the computer and then displayed on the image system. Using the image system, the experimenter can manipulate an image to enhance areas of interest for better analysis. Shortly after this system was placed in operation, it became apparent that viewing the data set in a one-to-one aspect ratio produced an image which was too small for detailed analysis. A cubic spline interpolation program was developed to uniformly expand the image to a larger format. Also, a two segment enhancement curve was developed to allow the operator to define his own video lookup table for pseudocolor translation of images stored in refresh memory.

SECTION II

DEVELOPMENT OF DATA ANALYSIS SYSTEM

This section of the report provides an overview of the display system and a description of its application to data analysis in pictorial format. The operation of the image display system is straight forward and does not involve any computer programming on the part of the user. A brief description of the commands used by the operator to interact with the system is presented.

A. Selection of a Display Media

The objective of this phase was to select a method for presenting experimental results in a manner that maximizes the information content and interpretability of the data. To date, the data display has primarily involved an isometric presentation that represents energy absorption or phase shift as a height above a planar surface whose perspective can be varied. The major disadvantage of this approach is that the aerial perspective contains no information. It was felt that an aerial perspective was critical to the successful analysis of the data because an aerial perspective would allow a determination of the spatial relation between the organ subdivisions.

The data obtained from the microwave scanner system is in a digital format. Each data point represents the energy absorption of a specific physical location on the organ. Using the co-ordinate information associated with each intensity value, the data can be assembled to form a mosaic "picture" of the energy absorption of the organ. This process is identical to that employed in reconstructing an image with data from a satellite. Hence, a literature search of commercial image processing systems was undertaken. Systems from the following vendors were considered:

- . Ramtek Corporation, Sunnyvale CA
- . Lexidata Corporation, Burlington MA
- . Grinnel Systems, San Jose CA
- . Comtal Corporation, Altadena CA

The Ramtek model 9351 Image Display System selected for this project for the following reasons:

- The modular architecture of the 9351 allowed the selection of a performance capability desired for this application. Also, field expansion or modi-

fication of the system at a later date would be easy to accomplish.

- The display refresh memory stores 16 bits per picture element.
- The function table provided for programmable definition of output intensity and color.
- The 9351 was on GSA schedule so it could be purchased at a substantial discount.
- Local service was available from the Ramtek office in Silver Spring, Maryland.
- An existing software package used for satellite image processing was available for the Ramtek which provided much of the processing necessary for display of the microwave data.

A price quotation was obtained from Ramtek to establish the price of the system. A letter proposal was then submitted to the sponsor recommending the Ramtek equipment and requesting additional funds for its procurement. While our proposal was being evaluated, two events occurred which were to impact this project. First, Ramtek instituted a 10 percent price increase on all their equipment and second, the GSA contract expired. Thus, when we received authorization to proceed with acquisition of the equipment, we were facing a 25 percent over-run in material cost. Over the next few weeks, we proceeded to negotiate an extension of the original price quotation from Ramtek in exchange for an extended delivery schedule.

Having defined the image processing system, the next task was to provide a hard copy capability. The most common method of making a hard copy at this time was to take a photograph of the color monitor. However, this required the services of a photographer and a darkroom capable of processing color prints. Not only was this a costly process but it was not well suited to a research environment where one is not able to predict when a hard-copy of the data will be required. In our search for an alternative, we contacted the Polaroid Corporation and were informed that they did not have a camera which would record the image directly. However, we were informed that two companies

- . Dunn Instruments, San Francisco CA
- . Matrix Instruments, Milan MI

made cameras for use with CAT scanners and used a newly developed 8½ by 11 Polaroid film. Subsequently, we were able to arrange for a demo of both camera systems. Each camera was fed an RGB video signal from the Ramtek and a number of photographs were taken.

The Matrix camera seemed more reliable and produced a slightly superior photograph. The Matrix camera also had a unique feature which allowed up to 16 images to be recorded on one 8½ by 11 sheet of film by reducing the size of each image. The capability of producing multiple images on one sheet of film was an extra cost option. Each time the number of images was doubled, the price increased by \$2000.00. In discussions with the technical officer, it was decided that two images per sheet of film was the optimum in price versus capability consistent with their needs at that time. The camera could be upgraded later by field modification if the need arose.

On March 19, 1980, a purchase order was issued to Matrix Corporation for a model 2001 camera system for delivery in 90 days. The first delivery date was not met and we were informed that production difficulties have caused the delivery schedule to slip. Many phone calls later, we were informed that Matrix had stopped production of the 2001 camera and was in the process of redesigning the entire system. The new camera, called a model 4007, was to incorporate all the optional features of the 2000 series camera plus have increased reliability and ease of operation. If we would agree to a six month delivery, Matrix proposed to deliver the new model for the same price as the model 2001. In discussions with the technical officer, it was decided that the six month delay was acceptable in return for a camera system with much improved performance. As a result of this change, the government received a camera system worth \$4000 more than the price they were charged. Unfortunately, the schedule on the new camera also slipped and it was not delivered until April 3, 1981. It should be noted that the Matrix camera has lived up to all the expectations and is considered well worth the extended time necessary to obtain it.

B. System Description

Figure 1 is a detailed block diagram of the RM-9351 System. The various subsystems are briefly described in the following paragraphs.

Host Parallel Link - The host parallel link provides the high speed (up to 600 KHz) bidirectional, 16-bit parallel communications between the host computer and the display system. Four external interrupts are implemented. The computer uses direct memory access (DMA) to communicate with the Ramtek GPIF, which is typically contained within the control/video board.

Internal Processor Bus - The internal processor bus connects the host parallel link, Z80 microprocessor, registers, memories, video generators, and interfaces for optional peripheral equipment. This bus provides high-speed device-to-device communication and implements nonprocessor DMA.

Display Address and Data Registers - Display address and data

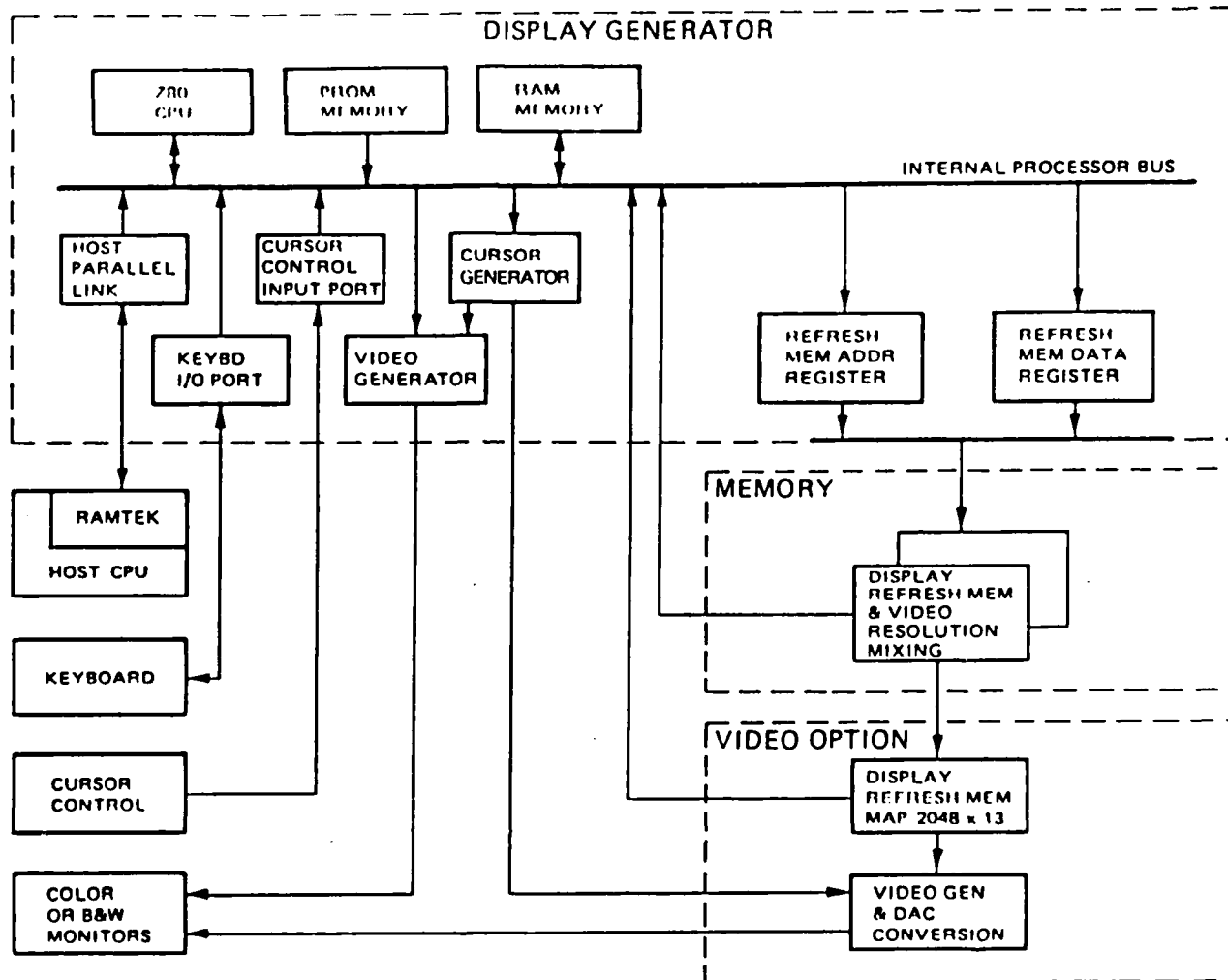


FIGURE 1

RM-9351 Image Display System

registers connect the bus and refresh memory for image generation and retrieval. This communication is in the form of a 16-bit word per pixel, with up to 16 bits being written in the z-axis. The X- and Y- addresses of the pixels are incremented as prescribed by the control registers and logic in the display generator. The display processor interface is optimized with respect to its internal algorithms for generating character and graphics data.

Graphic Display System - Graphics data is written under the direction of the current operating point (X-Y address), that is, up or down and/or left or right. The color or intensity of font, raster, and graphics data (Z-axis) is assigned by the Z80 microprocessor. Foreground, background, reversal, and writing mode (replacement versus additive) may be specified. The Z80 microprocessor is a powerful tool for implementing not only these modes of writing but any other application-dependent mode.

Video Monitor - The video monitor decodes the generated video signal and displays the image by driving one or more cathode-ray beams in raster fashion. Tube refresh time is at 30-Hz for monostatic systems.

Display Processor - This display processor interprets display-instruction information and presides over the bus. Secondary functions include character generation, vector generation, plot generation, raster mode, and raster margining. A Z80 microprocessor with 1,024 bytes random-access-memory (RAM) and 5,120 bytes of PROM is included. A basic instruction set is implemented, which provides imaging, graphics, and text-generating functions.

Refresh Memory - The refresh memory provides sixteen bits of storage for refreshing each pixel on the CRT. The refresh memory contains two memory boards, each board containing eight sections of memory. Each section contains sixteen 16K MOS RAMs that offer 512-line by 512-element refresh capability in the RM-9351 model. One plane of memory stores one bit of storage per pixel. For example, a refresh memory board in the RM-9351 contains sixteen sections of memory, i.e., one plane per section. Thus, the RM-9351 offers 16 planes or 16 bits of storage per pixel.

Video Generation - The video generation section of the control board can be modified through PROM coding to yield a multitude of possible configurations. Figure 2 shows the video generation network contained on the control/video board. By utilizing only the basic control/video board and one eight-plane memory board, the user can specify the PROM code so that the basic four video amps provide him with any of the following systems:

- a. Four planes (plane 0 through plane 3) patched to four video amps to yield four black-and-white displays. This is the standard configuration.
- b. Six planes patched to three video amps (plane 0

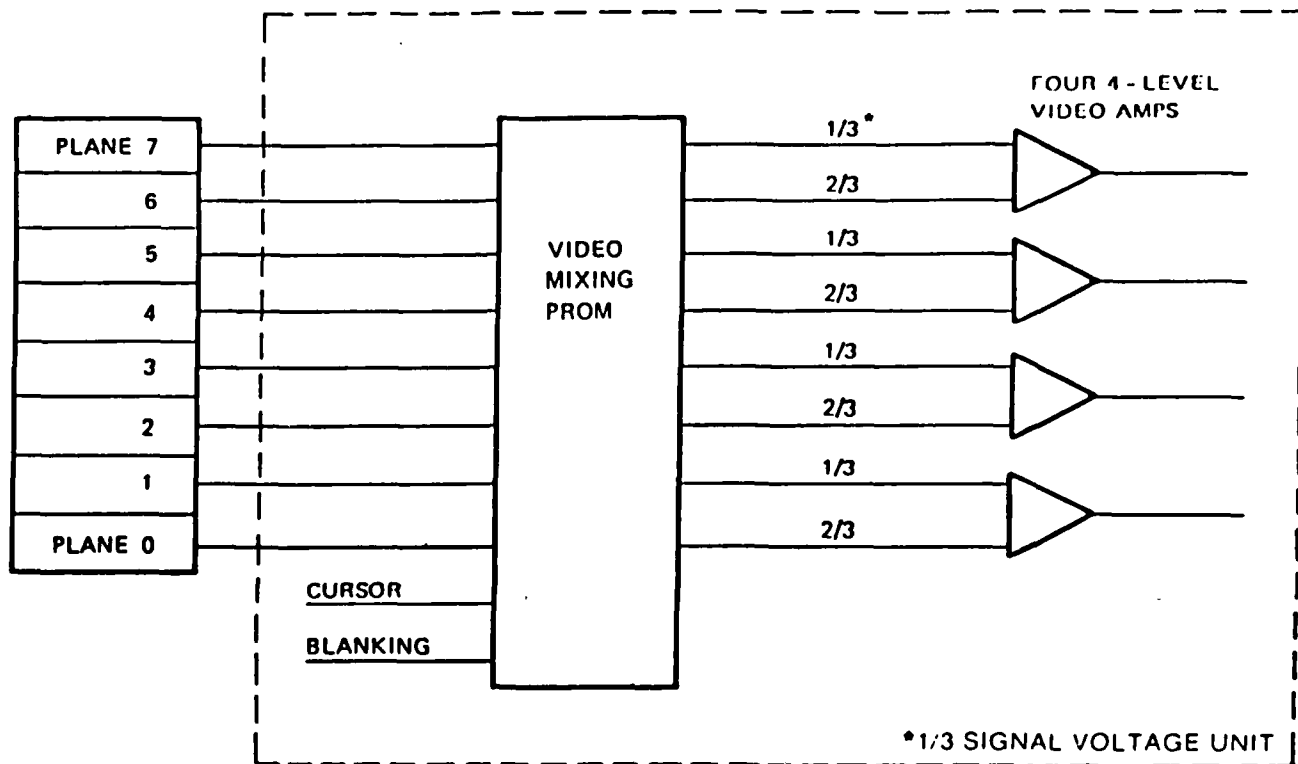


FIGURE 2

RM-9351 Control/Video Section

through plane 5) to yield 64 colors to drive an RGB monitor. The remaining video can be used to drive a composite signal of planes 0 through 5 to a B-W monitor which can provide four gray-scale levels.

- C. Eight planes patched to provide 64 colors on an RGB monitor (planes 0 through 5) with two planes provided to yield two overlays (planes 6 and 7).

The remaining video amp may drive, using planes 0 through 7, a composite black-and-white picture.

Enhancement tables are video lookup tables that allow interactive pseudocolor or gray-scale translation of images stored in refresh memory. True color results when a color monitor is driven directly by memory data; pseudocolor results when memory data is manipulated with a RAM before being sent to a color monitor. When equipped with a lookup table, the stored refresh data is treated as an address to the lookup table, which is host programmable; that is, as each pixel is scanned from the refresh memory for video presentation, the contents of the corresponding cell is retrieved from the lookup table and this data is passed to the digital-to-analog converters and video amplifiers, instead of the refresh data itself. Thus, the refresh data addresses a host programmable lookup table that assigns the output intensity or color.

Lookup tables are most often used in imaging applications. Figure 3 illustrates a commonly used enhancement procedure termed windowing, or density (level) slicing. Here, a specified range or group of contiguous image intensity values are fitted to the available spectrum of output intensity levels. Pixels having values beneath or above the specified range (or window) are translated to black or white, respectively; while pixels within the window are translated to an appropriate gray level. Thus, the observer's attention is focused upon the window, and he can more easily distinguish between what were minor or negligible intensity differences in the original image. More important, the user is able to manipulate this window interactively by reloading the lookup table, and without affecting the image in refresh memory. The integrity of the original image is thus maintained while the visual presentation is varied to suite the needs of the observer.

Figure 3 illustrates the effect of a relatively simple contrast enhancement algorithm. It is important to note that the equipment supports sophisticated algorithms. For example, a gamma corrected output can be achieved by loading a non-linear function into the lookup table, or a pre-defined or computed set of pseudocolors might be assigned to a gray scale image, etc.

The enhancement table consists of 2,048 cells of nine or thirteen bits. Thus, up to 11 bits of image data (in refresh memory)

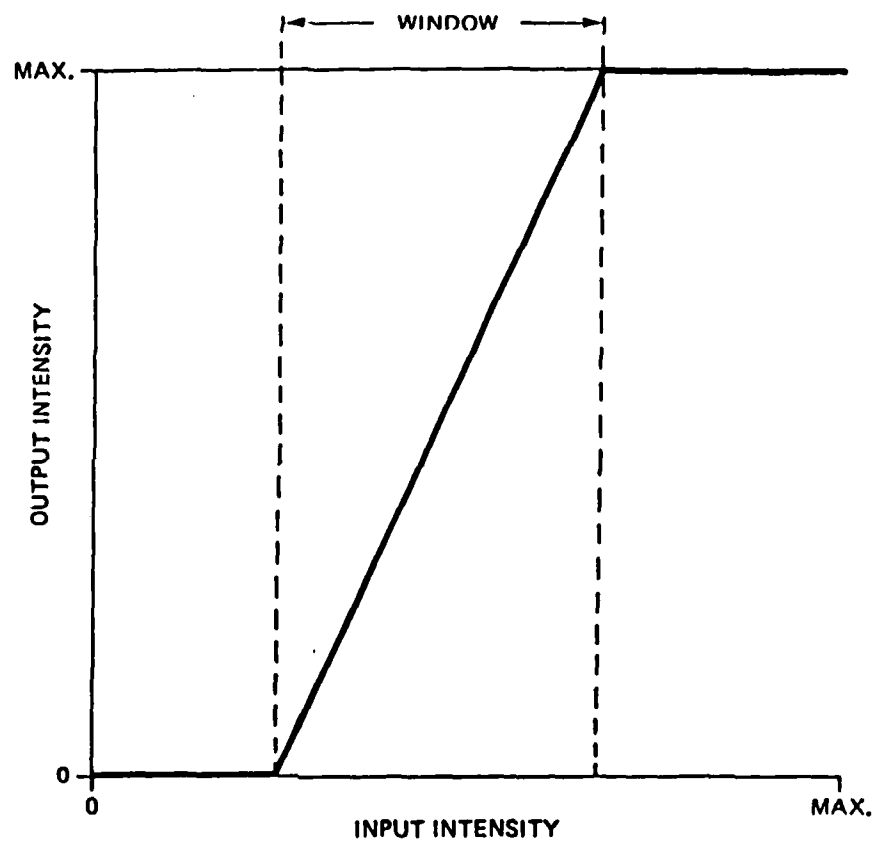


FIGURE 3

Example of Density Slicing

can address the table. For gray scale applications, bit 2₀ through 2₇ of the cells are output to a single 8-digital-to-analog converter (DAC). Thus any of 256 gray level intensities (I) may be produced, i.e., $I = N/255$ where $0 \leq N \leq 255$. For color applications, bits 2₀ through 2₃, 2₄ through 2₇, and 2₈ through 2₁₁ are output to three separate 4-bit DAC's corresponding to the red (R), green (G), and blue (B) primary inputs to an RGB color monitor. Thus, any of 4,096 colors (C) can be produced, i.e., $C = 2^8 \times R + 2^4 \times G + 2^0 \times B$ where $0 \leq R, G, B \leq 15$. Whether gray scale color, bit 2₁₂, when non-zero, causes the corresponding color or intensity to blink at a 2 Hz frequency.

C. Image System Operation

The operation of the image system was designed to provide the user with a very capable system that is simple to operate. All communication between the system and user takes place through the computer CRT console. A menu of the system commands is displayed on the CRT as an aid to the operator. To select a particular function, the operator need only type the first two characters of the particular command. The image system commands are listed below along with a brief description of their function.

STOP, END, OR EXIT

terminates program execution.

INITIALIZE

Program is initialized. All parameters are set to their default values. Next input command may now be chosen from list on CRT.

GRID

This gives the user the capability of varying either the picture or the grid intensity. The program will prompt the user with Input grid in (0 to 1). The user must input the desired grid brightness by punching in a number between 0.0 (black) and 1.0 (white). A number greater than 1.0 will cause the grid to become white and the brightness of the picture will decrease by an inversely proportionate amount. I.e., if a value of two is input the picture will become half as bright while the grid will become fully white. If a value of .5 is input the picture will be at maximum brightness while the grid will become half as bright. To return to ROLOR any of the input commands on the CRT may be selected.

TRANSFER

This command transfers a new picture from the disc to the RAMTEK display. The program will prompt the user with Picture disc Logical Unit. Punch in the LU number and press RETURN key. The program will then prompt "Picture number". Input the number of

the picture you want transferred and press RETURN key. The appropriate picture will be sent to display system. On the CRT the prompt picture number is returned in case another picture transfer is required. Any ROLOR's commands may also be input to return control to ROLOR and then to go to the specified command.

COLOR

This allows the user to select different color schemes for the pictures. The choice is varied and listed below. The program will prompt the user with choice (A through K, BW, A1 A9 or B1 B9). Choose a color scale (from the list below) and input your choice. The color schme of the picture will change at once. The input prompt will remain in case another color change is required. Any of ROLOR's commands may also be input to return control to ROLOR and then go to the specified command.

Key to color codes:

- A- Spectral (ROYGBIV)
- B- Half contouring of A
- C- Full contouring of A
- D- Green, Blue, Black, Green, Yellow, Red, Pink
- E- From Bull of Amer Meteorological Soc. Vol. 52, #9 (Sept 71)
- F- For NIMBUS 5 - ESMR pictures
- G- Yellow, Dark Orange, Dark Blue, Light Blue
- H- Blue, Green, Violet, Dark Orange White
- I- Blue, Green, Yellow, Red, Pink, White
- J- Variation of I
- K- Pink, Orange, Yellow, Green, Blue
- L- Microwave pseudocolor scale
- BW- Black and white
- A1 through A9- Contour intervals in green
 - A1 contours one interval (center of scale)
 - A2 contours two equally spaced intervals, etc.
 - Position of intervals contoured can be shifted by changing the color scale limits using the SC command

B1 through B9- Same as A1 through A9, except black and white

BACKGROUND

This changes the background color for the title and color bar. The background color is given by three values, each ranging from 0 to 1, which specify the intensities of red, green and blue respectively in the background.

e.g. 0,0,0 for black background
 0,1,0 for green background
 .4,0,.6 for purple background

when the program prompts Input red, green, blue (0 to 1) values then input your selection and press the RETURN key. The color

change is immediate on the RAMTEK display system. The prompt will reappear in case another color change is required. If not, any of ROLOR's input commands (listed on CRT) may be used. Control is returned to ROLOR which then executes the specified command.

BLOW-UP

This allows the user to enlarge a portion of the picture. The program will prompt the operator with "Picture disc Logical Unit." Input LU number of the picture disc. Press the RETURN key. The program will then prompt "Input line #, column #, factor, (Pict #)." The line # and the column # specify the center of the area that is to be enlarged. Values from 1 to 511 may be input for the two parameters. "Factor" is the amount by which the picture is to be enlarged. I.e., a factor of two will make the picture twice as large. The picture # is an optional parameter for the number of picture that is to be enlarged. If not specified it defaults to the value of the current picture. Input your selection and press RETURN key. The prompt specified above will reappear in case another change is needed. If not, any of ROLOR's commands may also be used. That command will then be executed.

DATA or SCALE

This provides the user with the ability to change the limits of the data scale and has a kaleidoscope and upkill facility. The user must supply four parameters, separated by commas, when the program gives the prompt "Input scaling for white 1/black = 0, upkill, klido"

- 1st para - Upper limit of data scale. If less than 1 the data with values beyond the limit will be displayed as white (or the current background color).
- 2nd para - Lower limit of data scale. If greater than 0 the data with values below the limit will be displayed as black.
- 3rd para - Upkill (0 or 1). If the value is 1 data with values above upper limit is displayed as black. If the value is 0 the data is white. It is an optional parameter and defaults to 0.
- 4th para - Kaleidoscope facility. This causes the picture to cycle through brightness scales automatically. The larger the number the slower the cycling. The length of time for which it runs is fixed and operator must wait for the whole sequence to end before issuing another command.
- Default values are 1,0,0, no kaleidoscope. The third parameter is optional. If not specified it equals 0. The fourth parameter is also optional. If not specified there is no kaleidoscope.

- If the lower limit is greater than the upper limit, the color scales are complimented.
- If an error occurs control is returned to ROLOR and you must specify the next input command from those listed on CRT.

ARCHIVE

This command references another program, program RATPT. It provides the user with the ability of transferring a complete 512 x 512 picture between the disc, the mag tape or the RAMTEK. To initiate a transfer, the operator must select one of the command options displayed on the CRT, type in the two letter mnemonic for the command, and press the RETURN key.

OVERLAY

This command references another program, program OVTPR. It provides the user with the capability of changing the grid and background colors of a picture to suit the picture requirements.

SECTION III

IMAGE PROCESSING PROGRAMS

A. CUBIC SPLINE INTERPOLATION

The microwave scanning system produces a 64 x 64 pixel image. The RAMTEK image system supports a 512 x 512 pixel image, so the raw image would occupy only one eighth of the screen, and little detail would be visible. There are numerous ways to enlarge an image, such as simple pixel replication or bilinear interpolation. While these methods are easier to implement than cubic spline interpolation, they have the disadvantage of degrading fine detail in the image. Thus, it is worth the effort to interpolate with cubic spline functions.

The particular method of cubic spline interpolation which was finally selected for implementation is outlined in Numerical Methods by Robert Hornback. This source describes how to fit a cubic spline to a set of points $(x_i, f(x_i))$. This is done as follows. Between each pair of adjacent points x_i and x_{i+1} it is necessary to find a cubic polynomial which passes through $(x_i, f(x_i))$ and $(x_{i+1}, f(x_{i+1}))$. This polynomial is denoted by $F_i(x) = a_0 + a_1x + a_2x^2 + a_3x^3$ for $x_i \leq x \leq x_{i+1}$. There are 4 unknown constants $F_i(x_{i+1}) = f(x_{i+1})$. The remaining constraints are imposed by requiring that the first and second derivatives of F_i match those of the polynomial F_{i-1} used on the previous interval. This gives the cubic spline interpolator its characteristic smoothness. To actually calculate the cubics F_i , it is first necessary to calculate the second derivatives $g''(x_i)$ which can be found by solving the following set of simultaneous equations:

$$\left[\frac{\Delta x_{i-1}}{\Delta x_i} \right] g''(x_{i-1}) + \left[\frac{2(x_{i+1} - x_{i-1})}{\Delta x_i} \right] g''(x_i) + g''(x_{i+1})$$

$$+ 6 \left[\frac{f(x_{i+1}) - f(x_i)}{(\Delta x_i)^2} - \frac{f(x_i) - f(x_{i-1}))}{(\Delta x_i)(\Delta x_{i-1})} \right] \quad (i = 1, 2, \dots, n-1)$$

where $\Delta x_i = x_{i+1} - x_i$.

Note that this procedure only $n-1$ equations in $n+1$ unknowns $g''(x_0), \dots, g''(x_n)$. The 2 additional equations are obtained by

specifying conditions on $g''(x_0)$ and $g''(x_n)$. For this purpose, take $g''(x_0) = 0$ and $g''(x_n) = 0$.

For image processing, consider the x_i to be at integer values and the functional values to be the pixel intensities. Thus $x_i = 1$, and the equations are simplified. The functions F_i can now be written in terms of g'' and f :

$$F_i(x) = \frac{g''(x_i)}{6} \left[(x_{i+1}-x)^3 - (x_{i+1}-x) \right] + \frac{g''(x_{i+1})}{6} \left[x - x_i \right]^3 - (x-x_i) + f(x_i) (x_{i+1}-x) + f(x_{i+1}) (x - x_i)$$

As an example, consider the effect of expanding a one dimensional edge which consists of five pixels with intensities 0,0,0, 512,512. In this case, $x_0 = 0, \dots, x_4 = 4$ and $f(x_0) = 0, \dots, f(x_4) = 512$.

We get the following conditions on g'' :

$$\begin{vmatrix} 4 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 4 \end{vmatrix} \begin{vmatrix} g''(1) \\ g''(2) \\ g''(3) \end{vmatrix} = 6 \begin{vmatrix} 0 \\ 512 \\ -512 \end{vmatrix}, \quad g''(0), g''(4) = 0$$

$$\text{hence } \begin{vmatrix} g''(1) \\ g''(2) \\ g''(3) \end{vmatrix} = \frac{6}{56} \begin{vmatrix} -5k \\ 12k \\ -19k \end{vmatrix} \quad \text{where } k = 512$$

and the equations are:

$$F_0(x) = \frac{-5k}{56} (x^3 - x)$$

$$F_1(x) = \frac{-5k}{56} \left[(2-x)^3 - (2-x) \right] = \frac{12}{56} \left[(x-1)^3 - (x-1) \right]$$

$$F_2(x) = \frac{12k}{56} \left[(3-x)^3 - (3-x) \right] - \frac{19k}{56} \left[(x-2)^3 - (x-2) \right] + k(x-2)$$

$$F_3(x) = -19k \left[(4-x)^3 - (4-x) \right] + k$$

To interpolate one pixel in between each original pixel we must compute $F_0(.5)$, $F_1(1.5)$, $F_2(2.5)$, and $F_3(3.5)$.

$$F_0(.5) = .033k = 17$$

$$F_1(1.5) = -.047k = -24 = 0 \quad (\text{for image processing})$$

$$F_2(2.5) = .547 = 280$$

$$F_3(3.5) = 1.127k = 577$$

The expanded edge is slightly enhanced and smoothed, as shown in Figure 4.

Since the cubic spline interpolator applies to one dimensional arrays of data and an image is two-dimensional, it is essentially necessary to make two passes with the interpolator. First, treat each column of the image as a set of data and interpolate the required number of pixels in each column. Then, treat each original plus interpolated row of the image as a one dimensional data set and interpolate horizontally.

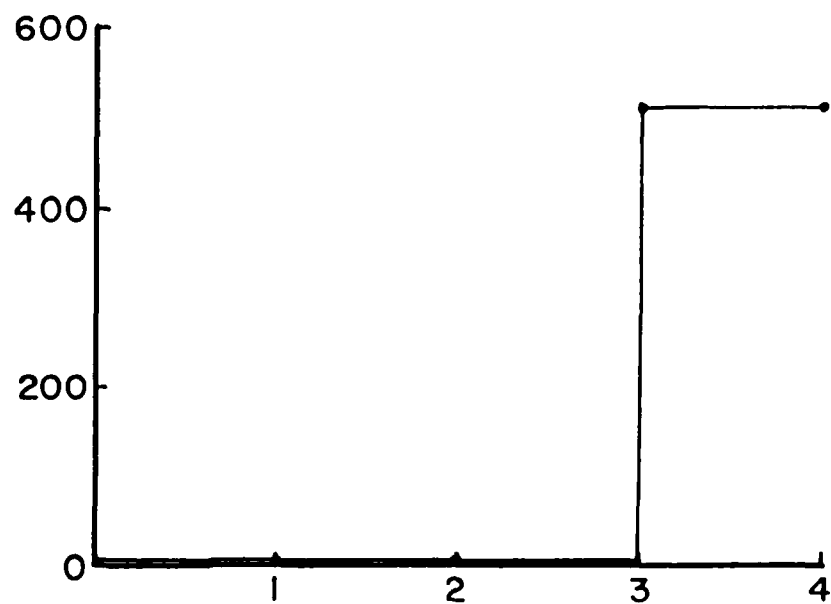
Conceptually, the method is shown in Figure 5.

In practice, to conserve memory requirements not all the vertical interpolation is done at once. Instead, all the information needed to construct the interpolating polynomials for each column is computed and stored. Then vertical interpolation is performed on two adjacent column pixels for every column, followed by horizontal interpolation across the original and interpolated rows. Then the data is flushed to the Ramtek and work begins on the next adjacent column pixels.

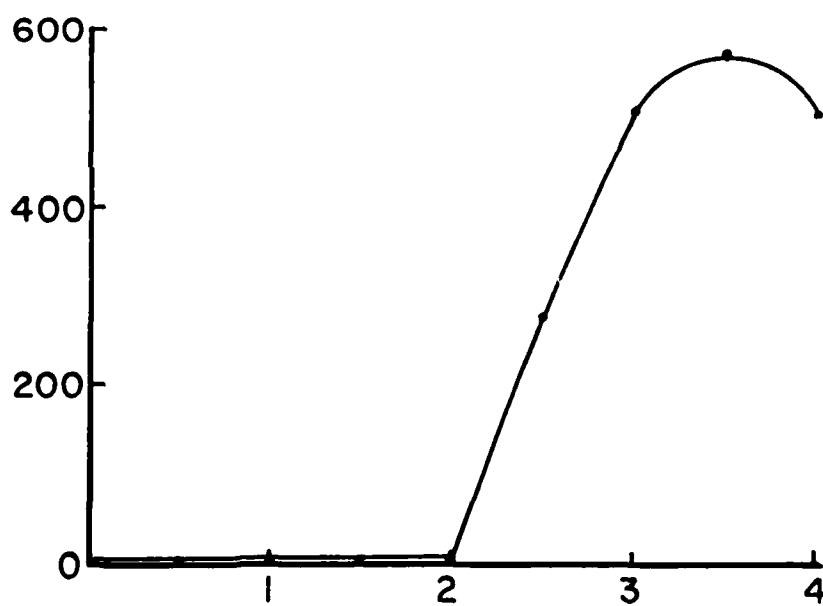
To allow the user maximum flexibility, this implementation of the cubic spline interpolation is not hard coded to any particular expansion factor. Due to screen size limitations it is not practical to insert more than 7 pixels between the originals, but the program can handle any expansion factor between 1 and 8, inclusive.

B. TWO SEGMENT ENHANCEMENT CURVE

The addition of a two segment enhancement curve to the RAMTEK 9351 system was accomplished by implementing a new subroutine and operator calls to the previously existing RAMTEK software package. The implementation allows a choice of single line or dual line enhancement curves by appending additional information to the enhancement operator query. Because the enhancement curve must be single valued, various tests are included to prevent the generation of meaningless curves. Internally, the software multiplexes the requested color scale and the enhancement curve to produce the video lookup table bit format, which is then transferred to the Image System. The enhancement curve with descriptions is shown in Figure 6.

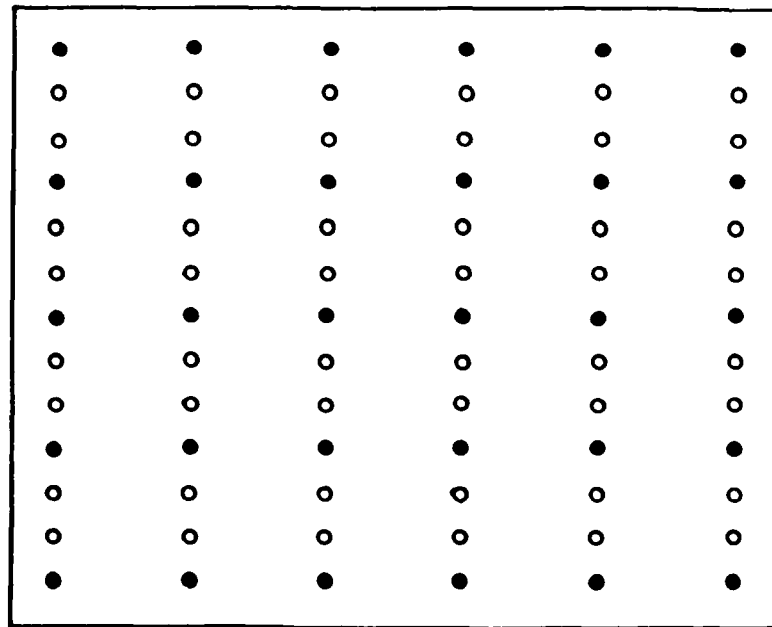


ORIGINAL

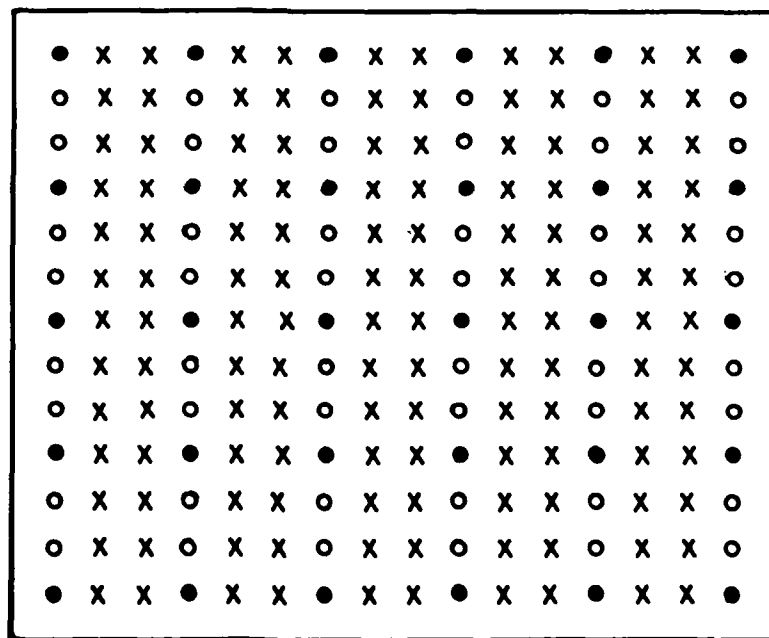


INTERPOLATED

FIGURE 4



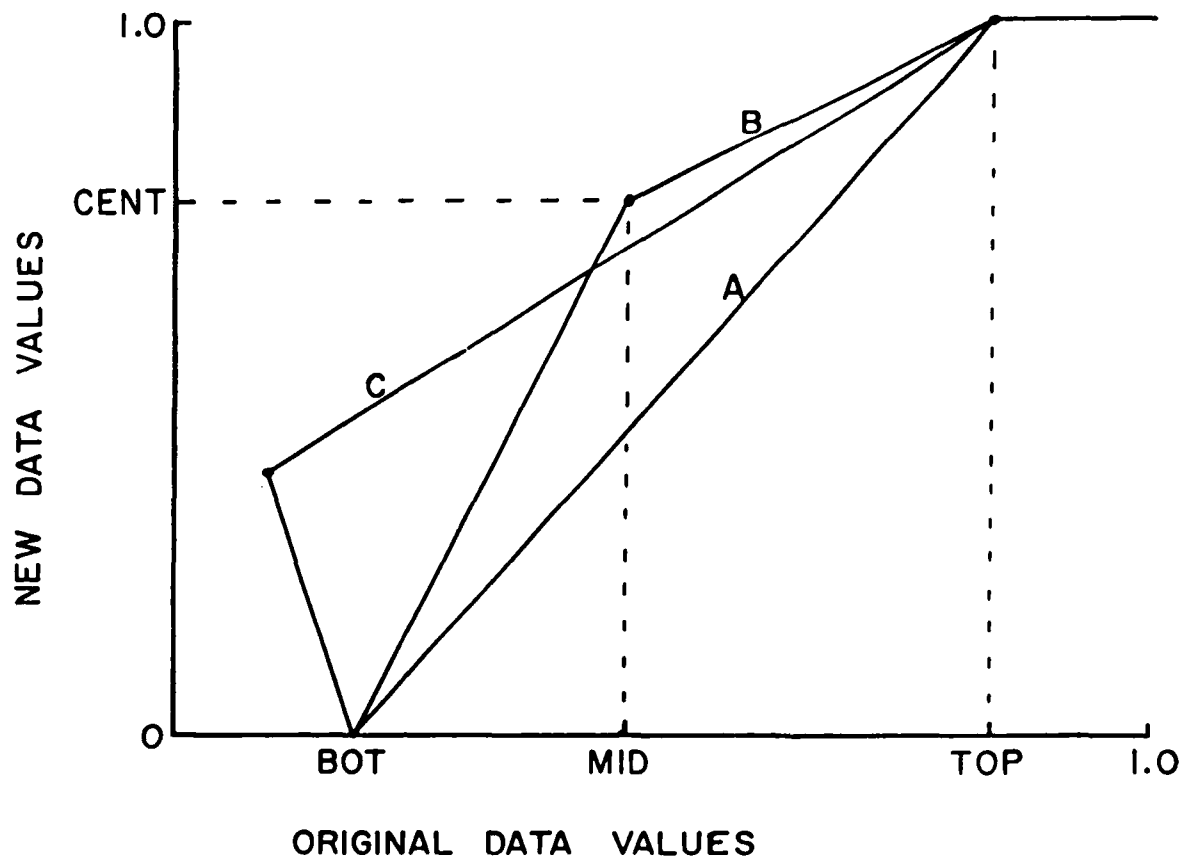
FIRST PASS



SECOND PASS

- Original Pixels
- Vertically Interpolated Pixels
- x Horizontally Interpolated Pixels

FIGURE 5



- A - Linear Enhancement Curve
- B - Two Segment Enhancement Curve
- C - Non-singular, Illegal Curve

FIGURE 6

SECTION IV

0001 FTN4,L

APPENDIX

```

0002 C*****
0003 C*          Written by Ginny Kalb          *
0004 C*          Technology U. S. A.          *
0005 C*          P. O. Box 55333 Ph (301) 292-2592 *
0006 C*          Oxon Hill Station          *
0007 C*          Washington D. C. 20022      *
0008 C*          Rev. 29 Jan 1981          *
0009 C*****
0010 C This program uniformly expands a square picture in both dim-
0011 C ensions by using a cubic spline interpolation. This is done
0012 C by first interpolating along the columns of the input picture,
0013 C then across the original plus interpolated rows. The resultant
0014 C image is displayed on the RAMTEK or COMTAL screen and optionally
0015 C stored on disc. Padding with zeros is used to obtain a full
0016 C screen output picture.
0017 C
0018 C NOTE: Several array sizes will need to be altered for differ-
0019 C ent input picture dimensions, as well as parameter IDIM =
0020 C pixels/side of input picture. The array sizes are a function
0021 C of IDIM and are located in COMMON blocks INPUT, UNIQ, and RROW.
0022 C Subroutine TREAD is of course input-unique and must be altered
0023 C for a new input picture size.
0024 C Array sizes in subroutine INVRS, used by SOLVE, must be changed
0025 C to A(IDIM-2,IDIM-2) and J(IDIM+19).
0026 C
0027 C COMMON /INPUT/IBUF(IDIM,IDIM)
0028 C COMMON /UNIQ/NPTS,IDIM,CCOEF(IDIM,4),DELTA,ISHF,XSTEP,
0029 C          DERIV(IDIM,IDIM-2),T(IDIM-2,IDIM-2)
0030 C COMMON /RROW/A(4),S(IDIM-2),Y(IDIM-2)
0031 C
0032 C The number of interpolated points is a user option at run time.
0033 C The default is 3 which expands a 64x64 picture to 253x253. The
0034 C maximum value for a 64x64 input picture is 7 because after that
0035 C the output picture would exceed 512x512.
0036 C*****
0037 CXY
0038 C
0039 C      PROGRAM EXPND
0040 C      COMMON /INPUT/IBUF(64,64)
0041 C      COMMON /OUTPT/IOPT,LUD,IPIC,L,JBUF(514),NPRT,LU3,ITK,ISECT
0042 C      COMMON /UNIQ/NPTS,IDIM,CCOEF(64,4),DELTA,ISHF,XSTEP,
0043 C*          DERIV(64,62),T(62,62)
0044 C      COMMON /RROW/A(4),S(62),Y(62)
0045 C      DIMENSION KBUF(512)
0046 C      EQUIVALENCE (KBUF(1),JBUF(3))
0047 C      DIMENSION IP(5)
0048 C
0049 C      CALL RMPAR(IP)
0050 C      WRITE(IP,1)
0051 C      1 FORMAT("EmEhEJ","E&a17rOC&dOEXPANSION VIA CUBIC SPLINE")
0052 C      WRITE(IP,2)
0053 C      2 FORMAT("E&a19rOC&K&dCNOTE&d@ program defaults to RAMTEK",
0054 C* " screen on LU 12.", "E&a20rOC&KDo you want to change either",
0055 C* " default? Y/N _")
0056 C      READ(IP,30) ITEMP
0057 C      IF(ITEMP.EQ.1HN) GO TO 6
0058 C      WRITE(IP,3)

```

```

0059      3 FORMAT("E&a20r0CEK", "E&a19r0CEKE&dC.E&d@input E&dCC0",
0060      * "E&d@ for CONTAL or E&dCRAE&d@ for RAMTEK _")
0061      READ(IP,4) NPRT
0062      4 FORMAT(A2)
0063      WRITE(IP,5)
0064      5 FORMAT("E&a20r0CEK", "E&a19r0CEKE&dC.E&d@input LU # _")
0065      READ(IP,*) LU3
0066      6 CONTINUE
0067      WRITE(IP,7)
0068      WRITE(IP,8)
0069      7 FORMAT("E&a19r0CEKE&dC.E&d@specify expansion factor --",
0070      * " e.g. 2 will double input picture which is 64x64 ")
0071      8 FORMAT("E&a20r0CEK(or just RETURN and use the default of 4)")
0072      ITEMP = -1
0073      READ(IP,*) ITEMP
0074      IF(ITEMP.GE.0) NPPTS = ITEMP-1
0075      IF(((IDIM-1)*NPPTS+IDIM).GT.512) GO TO 995
0076      WRITE(IP,10)
0077      10 FORMAT("E&a20r0CEK", "E&a19r0CEKE&dC.E&d@specify input ",
0078      * "picture tape unit _")
0079      READ(IP,*) MT
0080      C
0081      CALL INVRST,62,DTNRM,DETM)
0082      C
0083      15 WRITE(IP,20)
0084      20 FORMAT("E&a20r0CEK", "E&a19r0CEKE&dC.E&d@do you want ",
0085      * "to save output picture? Y/N _")
0086      READ(IP,30) IOPT
0087      30 FORMAT(A1)
0088      IF(IOPT.EQ.1HN) GO TO 50
0089      WRITE(IP,40)
0090      40 FORMAT("E&a19r0CEKE&dC.E&d@specify output disc LU and",
0091      * " picture number _")
0092      READ(IP,*) LUD,IPIC
0093      C
0094      C Verify that the specified output disc is legitimate
0095      C
0096      CALL CHECK(IP,LUD)
0097      C
0098      50 CONTINUE
0099      C
0100      C Initialization of parameters
0101      C
0102      L = -1
0103      ISHF = (512-((IDIM-1)*NPPTS+IDIM))/2
0104      XSTEP = (NPPTS+1.0)*DELTA
0105      ICNWD = MT+100B
0106      C
0107      C If using RAMTEK, issue a reset command
0108      C
0109      IF(NPRT.EQ.2HCO) GO TO 65
0110      CALL EXEC(2,LU3,2400B,1)
0111      C
0112      C Output leading rows of zero
0113      C
0114      65 LIM=(512-((IDIM-1)*NPPTS+IDIM))/2
0115      LDSK = (480-((IDIM-1)*NPPTS+IDIM))/2
0116      IF(NPRT.EQ.2HCO) LIM = LDSK
0117      IF(LIM.LE.0) GO TO 72
0118      DO 70 K=1,LIM

```

```

0119         IF(K.LE.LDSK) L = K-1
0120         CALL DSPLAY
0121         IF(IOPT.EQ.1HN) GO TO 70
0122         IF(K.GT.LDSK) GO TO 70
0123         CALL EXEC(2,LUD,KBUF,512,ITK,ISECT)
0124         70    CONTINUE
0125     C
0126     C Read in entire picture
0127     C
0128         72    CONTINUE
0129         DO 75 I=1,IDIM
0130             CALL TREAD(ICNWD,I,IP)
0131         75    CONTINUE
0132     C
0133     C Solve for column-determined second derivatives
0134     C
0135         DO 100 J=1,IDIM
0136             CALL SOLVE(0,J)
0137         100    CONTINUE
0138     C
0139     C Interpolate across first row
0140     C
0141         DO 210 J=1,IDIM
0142             INDEX = ISHF+(NPTS+1)*J-NPTS
0143             KBUF(INDEX) = IBUF(1,J)
0144         210    CONTINUE
0145             CALL ROW(1)
0146     C
0147     C Loop on remaining rows, doing column followed by row
0148     C interpolation
0149     C
0150         DO 350 I=2,IDIM
0151             N = I-1
0152             DO 310 J=1,IDIM
0153                 CALL GETCO(0,J,N)
0154             310    CONTINUE
0155     C
0156             IF(NPTS.EQ.0) GO TO 331
0157             DO 330 K=1,NPTS
0158                 X = (N-1)*XSTEP+K*DELTA
0159                 DO 320 J=1,IDIM
0160                     INDEX = ISHF+(NPTS+1)*J-NPTS
0161                     A(1) = CCOEF(J,1)
0162                     A(2) = CCOEF(J,2)
0163                     A(3) = CCOEF(J,3)
0164                     A(4) = CCOEF(J,4)
0165                     KBUF(INDEX) = F(A,X,N)
0166                     IBUF(I-1,J) = KBUF(INDEX)
0167             320    CONTINUE
0168                 CALL ROW(N)
0169             330    CONTINUE
0170             331    DO 340 J=1,IDIM
0171                 INDEX = ISHF+(NPTS+1)*J-NPTS
0172                 KBUF(INDEX) = IBUF(I,J)
0173             340    CONTINUE
0174                 CALL ROW(I)
0175             350    CONTINUE
0176     C
0177     C Write trailing rows of zero
0178     C

```

```

0179      DO 400 I=1,512
0180          KBUF(I) = 0
0181      400  CONTINUE
0182          LSAVE = L+1
0183      DO 410 L=LSAVE,511
0184          CALL DSPLAY
0185          IF(IOPT.EQ.1HN) GO TO 410
0186          IF(L.GT.479) GO TO 410
0187          CALL EXEC(2,LUD,KBUF,512,ITK,ISECT)
0188      410  CONTINUE
0189  C
0190      WRITE(IP,1)
0191      WRITE(IP,450)
0192      450  FORMAT("&a19r0C{Kit is finished}")
0193      WRITE(IP,455)
0194      455  FORMAT("&a20r0C{K{dC.{d@do you want to expand the ",
0195          * "next picture? Y/N _}")
0196      READ(IP,30) ITEMP
0197      IF(ITEMP.EQ.1HN) STOP
0198      ICNWD = MT+1300B
0199      CALL EXEC(3,ICNWD)
0200      GO TO 15
0201  C
0202      995  WRITE(IP,996)
0203      996  FORMAT("&a19r0C{Kinvalid value for expansion factor",
0204          * "&a20r0C{KTry a value between 1 and 8, inclusive")
0205      STOP
0206      END
0207  C
0208  C * * * * *
0209  C
0210      SUBROUTINE ROW(I)
0211  C
0212  C This subroutine interpolates NPTS points between the gray levels
0213  C in row I of the input buffer IBUF. This is either the original
0214  C row or an interpolated-between-columns row.
0215  C The resultant row is then displayed and optionally stored on disc.
0216  C
0217      COMMON /OUTPT/IOPT,LUD,IPIC,L,JBUF(514),NPRT,LU3,ITK,ISECT
0218      COMMON /UNIQ/NPTS,IDIM,CCOEF(64,4),DELTA,ISHF,XSTEP,
0219          * DERIV(64,62),T(62,62)
0220      COMMON /RROW/A(4),S(62),Y(62)
0221      DIMENSION KBUF(512)
0222      EQUIVALENCE (KBUF(1),JBUF(3))
0223  C
0224      IF(NPTS.EQ.0) GO TO 201
0225      CALL SOLVE(I,0)
0226      LIM = IDIM-1
0227      DO 200 J=1,LIM
0228          CALL GETCO(I,0,J)
0229          DO 100 K=1,NPTS
0230              X=(J-1)*XSTEP+K*DELTA
0231              INDEX = ISHF+(NPTS+1)*J-NPTS+K
0232              KBUF(INDEX) = F(A,X,J)
0233      100  CONTINUE
0234      200  CONTINUE
0235      201  L = L+1
0236      CALL DSPLAY
0237      IF(IOPT.EQ.1HN) RETURN
0238      IF(L.GT.479) RETURN

```

```

0239      CALL EXEC(2,LUD,KBUF,512,ITK,ISECT)
0240      RETURN
0241      END
0242  C
0243  C * * * * *
0244  C
0245      SUBROUTINE SOLVE(I,J)
0246  C
0247  C This subroutine solves the IDIM-2 simultaneous equations
0248  C for the second derivatives determined by the data values
0249  C in row I or column J of the input buffer IBUF.
0250  C
0251      COMMON /UNIQ/NPTS,IDIM,CCDEF(64,4),DELTA,ISHF,XSTEP,
0252      *      DERIV(64,62),T(62,62)
0253      COMMON /INPUT/IBUF(64,64)
0254      COMMON /RROW/A(4),S(62),Y(62)
0255  C
0256      LL = IDIM-2
0257      IF(I.EQ.0) GO TO 200
0258  C
0259  C Compute second derivatives for row I of input buffer.
0260  C
0261      DO 50 K=1,LL
0262          Y(K) = IBUF(I,K+2)-2*IBUF(I,K+1)+IBUF(I,K)
0263          Y(K) = Y(K)*6/(XSTEP*XSTEP)
0264      50  CONTINUE
0265  C
0266      DO 100 K=1,LL
0267          S(K) = 0.0
0268          DO 75 M=1,LL
0269              S(K) = S(K)+T(K,M)*Y(M)
0270      75  CONTINUE
0271      100 CONTINUE
0272      RETURN
0273  C
0274  C Compute second derivatives for column J of input buffer.
0275  C
0276      200 CONTINUE
0277      DO 250 K=1,LL
0278          Y(K) = IBUF(K+2,J)-2*IBUF(K+1,J)+IBUF(K,J)
0279          Y(K) = Y(K)*6/(XSTEP*XSTEP)
0280      250 CONTINUE
0281  C
0282      DO 300 K=1,LL
0283          DERIV(J,K) = 0.0
0284          DO 275 M=1,LL
0285              DERIV(J,K) = DERIV(J,K)+T(K,M)*Y(M)
0286      275 CONTINUE
0287      300 CONTINUE
0288      RETURN
0289      END
0290  C
0291  C * * * * *
0292  C
0293      SUBROUTINE GETCO(I,J,N)
0294  C
0295  C This subroutine computes coefficients needed to interpolate
0296  C span N along row I or column J.
0297  C
0298      COMMON /UNIQ/NPTS,IDIM,CCDEF(64,4),DELTA,ISHF,XSTEP,

```

```

0299      *          DERIV(64,62),T(62,62)
0300      COMMON /INPUT/IBUF(64,64)
0301      COMMON /RROW/A(4),S(62),Y(62)
0302  C
0303      IF(I.EQ.0) GO TO 100
0304  C
0305  C Row interpolation
0306  C
0307      A(1) = 0.0
0308      A(2) = 0.0
0309      IF(N.GT.1) A(1) = S(N-1)/6.0
0310      IF(N.LT.(IDIM-1)) A(2) = S(N)/6.0
0311      A(3) = IBUF(I,N)
0312      A(4) = IBUF(I,N+1)
0313      RETURN
0314  C
0315  C Column interpolation
0316  C
0317      100 CONTINUE
0318      CCOEF(J,1) = 0.0
0319      CCOEF(J,2) = 0.0
0320      IF(N.GT.1) CCOEF(J,1) = DERIV(J,N-1)/6.0
0321      IF(N.LT.63) CCOEF(J,2) = DERIV(J,N)/6.0
0322      CCOEF(J,3) = IBUF(N,J)
0323      CCOEF(J,4) = IBUF(N+1,J)
0324      RETURN
0325      END
0326  C
0327  C * * * * *
0328  C
0329      FUNCTION F(A,X,I)
0330  C
0331  C This routine computes the interpolated value for x coordinate X
0332  C which lies in the Ith span.
0333  C
0334      COMMON /UNIQ/NPTS,IDIM,CCOEF(64,4),DELTA,ISHF,XSTEP,
0335      *          DERIV(64,62),T(62,62)
0336      DIMENSION A(4)
0337      H = XSTEP
0338      DX1 = X-(I-1)*H
0339      DX2 = H-DX1
0340      F = A(1)*DX2*(DX2*DX2/H-H)
0341      *      +A(2)*DX1*(DX1*DX1/H-H)
0342      *      +A(3)*DX2/H+A(4)*DX1/H
0343      IF(F.LT.0.0) F=0.0
0344      RETURN
0345      END
0346  C
0347  C * * * * *
0348  C
0349      SUBROUTINE TREAD(ICNWD,I,IP)
0350  C
0351  C This subroutine reads one row of the input picture,
0352  C assuming the following format:
0353  C   1 row = 32 16-bit words = 64 8-bit pixels
0354  C
0355      COMMON /INPUT/IBUF(64,64)
0356      COMMON /UNIQ/NPTS,IDIM,CCOEF(64,4),DELTA,ISHF,XSTEP,
0357      *          DERIV(64,62),T(62,62)
0358      DIMENSION LBUF(32),IR(2),IP(5)

```

```

0359      EQUIVALENCE (IR,REG)
0360      REG = EXEC(1,ICNWD,LBUF,32)
0361      IF(IAND(IR,200B).EQ.200B) GO TO 995
0362      LL = IDIM/2
0363      IF(IR(2).GE.LL) GO TO 75
0364      N = IR(2)+1
0365      DO 50 K=N,LL
0366      50   LBUF(K) = 0
0367      C
0368      75 CONTINUE
0369      DO 100 J=1,LL
0370          IBUF(I,2*J-1) = IAND(LBUF(J)/256,377B)*4
0371          IBUF(I,2*J) = IAND(LBUF(J),377B)*4
0372      100 CONTINUE
0373      RETURN
0374      C
0375      995 WRITE(IP,996)
0376      996 FORMAT("E&a19r0C{KInvalid # of records in picture file}")
0377      STOP
0378      END
0379      C
0380      C * * * * *
0381      C
0382      SUBROUTINE DSPLAY
0383      C
0384      C This subroutine makes the appropriate EXEC call to write a line of
0385      C the output picture to the selected screen, specified by NPRT and LU3.
0386      C
0387      COMMON /OUTPT/IOPT,LUD,IPIC,L,JBUF(514),NPRT,LU3,ITK,ISECT
0388      DIMENSION KBUF(512)
0389      EQUIVALENCE (KBUF(1),JBUF(3))
0390      C
0391      ITK = L/12+(IPIC-1)*40
0392      ISECT = (L-(ITK-(IPIC-1)*40)*12)*8
0393      IF(NPRT.EQ.2HRA) GO TO 200
0394      C
0395      CALL EXEC(2,LU3,KBUF,512,L)
0396      RETURN
0397      C
0398      200 CALL EXEC(2,LU3,JBUF,514)
0399      RETURN
0400      C&Z
0401      END
0402      C
0403      C * * * * *
0404      C
0405      BLOCK DATA
0406      COMMON /INPUT/IBUF(64,64)
0407      COMMON /OUTPT/IOPT,LUD,IPIC,L,JBUF(514),NPRT,LU3,ITK,ISECT
0408      COMMON /UNIQ/NPTS,IDIM,CCOEF(64,4),DELTA,ISHF,XSTEP,
0409      *      DERIV(64,62),T(62,62)
0410      COMMON /RROW/A(4),S(62),Y(62)
0411      DATA NPTS/3/,IDIM/64/,T/0.0/,DELTA/0.5/
0412      DATA NPRT/2HRA/,LU3/12/
0413      DATA JBUF(1)/5001B/,JBUF(2)/1024/
0414      DATA T(1,1)/4.0/,T(1,2)/1.0/,
0415      *      T(2,1)/1.0/,T(2,2)/4.0/,T(2,3)/1.0/,
0416      *      T(3,2)/1.0/,T(3,3)/4.0/,T(3,4)/1.0/,
0417      *      T(4,3)/1.0/,T(4,4)/4.0/,T(4,5)/1.0/,
0418      *      T(5,4)/1.0/,T(5,5)/4.0/,T(5,6)/1.0/,

```

0419	*	T(6,5)/1.0/,T(6,6)/4.0/,T(6,7)/1.0/,
0420	*	T(7,6)/1.0/,T(7,7)/4.0/,T(7,8)/1.0/,
0421	*	T(8,7)/1.0/,T(8,8)/4.0/,T(8,9)/1.0/,
0422	*	T(9,1)/1.0/,T(9,9)/4.0/,T(9,10)/1.0/,
0423	*	T(10,9)/1.0/,T(10,10)/4.0/,T(10,11)/1.0/,
0424	*	T(11,10)/1.0/,T(11,11)/4.0/,T(11,12)/1.0/,
0425	*	T(12,11)/1.0/,T(12,12)/4.0/,T(12,13)/1.0/,
0426	*	T(13,12)/1.0/,T(13,13)/4.0/,T(13,14)/1.0/,
0427	*	T(14,13)/1.0/,T(14,14)/4.0/,T(14,15)/1.0/,
0428	*	T(15,14)/1.0/,T(15,15)/4.0/,T(15,16)/1.0/,
0429	*	T(16,15)/1.0/,T(16,16)/4.0/,T(16,17)/1.0/,
0430	*	T(17,16)/1.0/,T(17,17)/4.0/,T(17,18)/1.0/,
0431	*	T(18,17)/1.0/,T(18,18)/4.0/,T(18,19)/1.0/,
0432	*	T(19,18)/1.0/,T(19,19)/4.0/,T(19,20)/1.0/,
0433	*	T(20,19)/1.0/,T(20,20)/4.0/,T(20,21)/1.0/,
0434	*	T(21,20)/1.0/,T(21,21)/4.0/,T(21,22)/1.0/,
0435	*	T(22,21)/1.0/,T(22,22)/4.0/,T(22,23)/1.0/,
0436	*	T(23,22)/1.0/,T(23,23)/4.0/,T(23,24)/1.0/,
0437	*	T(24,23)/1.0/,T(24,24)/4.0/,T(24,25)/1.0/,
0438	*	T(25,24)/1.0/,T(25,25)/4.0/,T(25,26)/1.0/,
0439	*	T(26,25)/1.0/,T(26,26)/4.0/,T(26,27)/1.0/,
0440	*	T(27,26)/1.0/,T(27,27)/4.0/,T(27,28)/1.0/,
0441	*	T(28,27)/1.0/,T(28,28)/4.0/,T(28,29)/1.0/,
0442	*	T(29,28)/1.0/,T(29,29)/4.0/,T(29,28)/1.0/,
0443	*	T(30,29)/1.0/,T(30,30)/4.0/,T(30,31)/1.0/,
0444	*	T(31,31)/1.0/,T(31,31)/4.0/,T(31,32)/1.0/,
0445	*	T(32,31)/1.0/,T(32,32)/4.0/,T(32,33)/1.0/,
0446	*	T(33,32)/1.0/,T(33,33)/4.0/,T(33,34)/1.0/,
0447	*	T(34,33)/1.0/,T(34,34)/4.0/,T(34,35)/1.0/,
0448	*	T(35,34)/1.0/,T(35,35)/4.0/,T(35,36)/1.0/,
0449	*	T(36,35)/1.0/,T(36,36)/4.0/,T(36,37)/1.0/,
0450	*	T(37,36)/1.0/,T(37,37)/4.0/,T(37,38)/1.0/,
0451	*	T(38,37)/1.0/,T(38,38)/4.0/,T(38,39)/1.0/,
0452	*	T(39,38)/1.0/,T(39,39)/4.0/,T(39,40)/1.0/,
0453	*	T(40,39)/1.0/,T(40,40)/4.0/,T(40,41)/1.0/,
0454	*	T(41,40)/1.0/,T(41,41)/4.0/,T(41,42)/1.0/,
0455	*	T(42,41)/1.0/,T(42,42)/4.0/,T(42,43)/1.0/,
0456	*	T(43,42)/1.0/,T(43,43)/4.0/,T(43,44)/1.0/,
0457	*	T(44,43)/1.0/,T(44,44)/4.0/,T(44,45)/1.0/,
0458	*	T(45,44)/1.0/,T(45,45)/4.0/,T(45,46)/1.0/,
0459	*	T(46,45)/1.0/,T(46,46)/4.0/,T(46,47)/1.0/,
0460	*	T(47,46)/1.0/,T(47,47)/4.0/,T(47,48)/1.0/,
0461	*	T(48,47)/1.0/,T(48,48)/4.0/,T(48,49)/1.0/,
0462	*	T(49,48)/1.0/,T(49,49)/4.0/,T(49,50)/1.0/,
0463	*	T(50,49)/1.0/,T(50,50)/4.0/,T(50,51)/1.0/,
0464	*	T(51,50)/1.0/,T(51,51)/4.0/,T(51,52)/1.0/,
0465	*	T(52,51)/1.0/,T(52,52)/4.0/,T(52,53)/1.0/,
0466	*	T(53,52)/1.0/,T(53,53)/4.0/,T(53,54)/1.0/,
0467	*	T(54,53)/1.0/,T(54,54)/4.0/,T(54,55)/1.0/,
0468	*	T(55,54)/1.0/,T(55,55)/4.0/,T(55,56)/1.0/,
0469	*	T(56,55)/1.0/,T(56,56)/4.0/,T(56,57)/1.0/,
0470	*	T(57,56)/1.0/,T(57,57)/4.0/,T(57,58)/1.0/,
0471	*	T(58,57)/1.0/,T(58,58)/4.0/,T(58,59)/1.0/,
0472	*	T(59,58)/1.0/,T(59,59)/4.0/,T(59,60)/1.0/,
0473	*	T(60,59)/1.0/,T(60,60)/4.0/,T(60,61)/1.0/,
0474	*	T(61,60)/1.0/,T(61,61)/4.0/,T(61,62)/1.0/,
0475	*	T(62,61)/1.0/,T(62,62)/4.0/
0476	END	
0477	END*	

```

0001
0002 C*****
0003 C*      Written by Ginny Kalb      *
0004 C*      Technology U. S. A.      *
0005 C*      P. O. Box 55333 Ph (301) 292-2592 *
0006 C*      Oxon Hill Station      *
0007 C*      Washington D. C. 20022 *
0008 C*      Rev. 29 Jan 1981      *
0009 C*****
0010 C*      This program is used to document Program EXPND *
0011 C*****
0012 C
0013 01 Software Blueprint -- Level B Design of the Cubic Spline Expansion
0014 02 Program Modules
0015 03 Module Declaration
0016
0017      #1 10 MAIN          main program
0018      #2 20 INVRS         matrix inversion
0019      #3 20 TREAD         read one row of input tape
0020      #4 20 SOLVE         solve matrix eqs for 2nd derivatives
0021      #5 20 GETCO         compute interpolation coefficients
0022      #6 20 ROW           interpolate across row of data points
0023      #7 20 F             evaluate interpolated value
0024      #8 20 DSPLAY        output interpolated row to screen
0025 03 Module Reference Structure
0026      <calling proc name>::=<called proc name list>
0027
0028      #1 10 MAIN::=INVRS,DSPLAY,TREAD,SOLVE,ROW,GETCO,F
0029      #2 20 INVRS
0030      #3 20 TREAD
0031      #4 20 SOLVE
0032      #5 20 GETCO
0033      #6 20 ROW::=SOLVE,GETCO,F,DSPLAY
0034      #7 20 F
0035      #8 20 DSPLAY
0036 02 Data
0037 03 Data Declaration -- all names follow FORTRAN default data types
0038
0039      Buffer
0040      DELTA, relative spacing between adjacent output pixels
0041      ICNWD, control word for EXEC calls to tape drive MT
0042      IDIM, # pixels per side of input picture
0043      INDEX, pointer into output array KBUF
0044      IPIC, output picture number
0045      ISECT, sector number for storing picture on disc
0046      ISHF, bias count needed to center output picture row in
0047      array KBUF
0048      ITK, track number for storing picture on disc
0049      L, line count for COMTAL or disc output control
0050      LDISK, # of extra rows on top & bottom of screen when
0051      output picture is centered
0052      LIM, # of extra rows on top & bottom of disc space
0053      when output picture is centered
0054      LUD, logical unit number of output disc
0055      LU3, logical unit number of output screen
0056      Initialize LU3 to 12
0057      MT, logical unit number of input tape
0058      NPTS, # of data points to be inserted between adjacent

```

```

0059         input data pixels
0060         Initialize NPTS to 3
0061         X,      position coordinate of interpolated pixel
0062         XSTEP,  relative spacing between adjacent input pixels
0063
0064     Array
0065         A(4),      coefficients needed to interpolate between 2
0066                   pixels in a row
0067         CCOEF(IDIM,4), for each column, coefficients needed to
0068                   interpolate between 2 pixels in that column
0069         DERIV(IDIM,IDIM-2), for each column, 2nd derivatives at each
0070                   input pixel (assuming 0 at first and last
0071                   pixels)
0072         IBUF(IDIM,IDIM), stores entire input picture
0073         IP(5),      stores RMPAR parameters -- only first one is
0074                   used; IP = logical unit # of user's terminal
0075         JBUF(514),  output array for RAMTEK; first 2 words are
0076                   predetermined, rest are data words
0077                   Initialize JBUF(1) to octal 5001 and JBUF(2)
0078                   to decimal 1024
0079         KBUF(512),  output array for COMTAL or disc
0080         S(IDIM-2),  2nd derivatives at each input pixel in a row
0081                   (assuming 0 at first and last pixels)
0082         T(IDIM-2,IDIM-2) inverse of matrix arising from the IDIM-2
0083                   simultaneous equations for the 2nd derivativ
0084                   for a cubic spline fit through IDIM equally
0085                   spaced data points with 2nd derivative of 0
0086                   at the endpoints
0087                   Initialize T to 4 along the diagonal, 1
0088                   off the diagonal, and 0 elsewhere
0089
0090 03 Data Reference Structure
0091     <data structure type><data element name>::=<referencing proc's>
0092
0093     Buffer
0094         DELTA ::= MAIN,ROW
0095         ICNWD ::= MAIN,TREAD
0096         IDIM  ::= MAIN,TREAD,SOLVE,GETCO,ROW
0097         INDEX ::= MAIN,ROW
0098         IPIC  ::= MAIN,DSPLAY
0099         ISECT ::= MAIN,ROW,DSPLAY
0100         ISHF  ::= MAIN,ROW
0101         ITK   ::= MAIN,ROW,DSPLAY
0102         L     ::= MAIN,ROW,DSPLAY
0103         LDISK ::= MAIN
0104         LIM   ::= MAIN
0105         LUD   ::= MAIN,ROW
0106         LU3   ::= MAIN,DSPLAY
0107         MT    ::= MAIN
0108         NPTS  ::= MAIN,ROW
0109         X     ::= MAIN,ROW,F
0110         XSTEP ::= MAIN,SOLVE,ROW,F
0111
0112     Array
0113         A      ::= MAIN,GETCO,ROW,F
0114         CCOEF  ::= MAIN,GETCO
0115         DERIV  ::= SOLVE,GETCO
0116         IBUF   ::= MAIN,TREAD,SOLVE,GETCO
0117         IP     ::= MAIN,CHECK,TREAD
0118         JBUF   ::= DSPLAY
0119         KBUF   ::= MAIN,ROW

```

```

0119          S      == SOLVE,GETCO
0120          T      == INVRS,SOLVE
0121      02.Control
0122          03 Control Declaration
0123
0124          Switch
0125              IOPT Of Status ("Y","N")    "save on disc" response
0126              NPRT Of Status ("RA","CO") output screen identifier
0127      03.Control Reference Structure
0128          <switch.name>==<where-set>/<where tested>
0129          IOPT == MAIN/MAIN,ROW
0130          NPRT == MAIN/MAIN,DSPLAY
0131      02 Procedure Definition
0132          #1 Procedure MAIN
0133
0134          Questions asked of user at run time:
0135          (1) "program defaults to RAMTEK screen on LU 12.
0136              Do you want to change either default? Y/N"
0137          If answer is "Y", (1a) "input CO for COMTAL or RA for RAMTEK"
0138              (1b) "input LU #"
0139          (2) "specify expansion factor -- e.g. 2 will approximately double
0140              input picture which is 64x64. (or just RETURN and use default
0141              of 4"
0142          (3) "specify input picture tape unit"
0143          (4) "do you want to save output picture? Y/N"
0144          If answer is "Y", (4a) "specify output disk LU and picture number"
0145          At completion of expansion,
0146          (5) "do you want to expand the next picture? Y/N"
0147          Call RMPAR;          get LU # of user's terminal
0148          Write questions;    prompt user for specifics
0149          Read answers;
0150          Call INVRS;          invert matrix T
0151          Initialize L,ISHF,XSTEP, and ICNWD;
0152          If NPRT = "RA"
0153              Then issue reset command to RAMTEK;
0154          End If;
0155          Write leading rows of 0 to screen;
0156          If IOPT = "Y"
0157              Then write leading rows of 0 to disc;
0158          End If;
0159          Loop until entire input picture has been read
0160              Call TREAD;          read next row of input picture
0161          End Loop;
0162
0163          Loop until all columns of input picture have been processed
0164              Call SOLVE;          get 2nd derivatives at column pixels
0165                                  (<0 at endpoints)
0166          End Loop;
0167
0168          Embed 1st row of input picture in output array KBUF,
0169          leaving NPTS gaps in between for interpolated values;
0170          Call ROW;          fill in this row
0171
0172          Loop until all input rows have been processed
0173              Loop until all input columns have been processed
0174                  Call GETCO;          get coefficients to interpolate along this
0175                                          column between old and current input rows
0176              End Loop;
0177              If NPTS is not 0 Then
0178                  Loop until NPTS pixel positions have been processed

```

```

0179      X := X+DELTA; put X at next pixel position
0180      Loop until all input columns have been processed
0181      Call F; interpolate at X in this column
0182      Write new value in output array KBUF;
0183      Replace the data point in the input array IBUF at
0184      this column and old row by the new value;
0185      End Loop;
0186      Call ROW; interpolate across the row just manufactured
0187      Move current row pointer into old row pointer;
0188      End Loop;
0189      End If;
0190      Embed current input row in output array KBUF, leaving
0191      NPTS gaps in between for interpolated values;
0192      Call ROW; fill in this row
0193      End Loop;
0194      Write trailing rows of 0 to screen;
0195      If IOPT = "Y"
0196      Then write trailing rows of 0 to disc;
0197      End If;
0198      End MAIN;
0199

```

#2 Procedure INVRS

```

0200      This subroutine was obtained verbatim from NUMERICAL METHODS
0201      by Robert W. Hornbeck.
0202      This subroutine is only called at initialization to invert
0203      matrix T.
0204
0205      Array
0206      A(IDIM-2, IDIM-2), original matrix
0207      J(IDIM-2+21) temporary storage
0208      Buffer
0209      M dimension of matrix to be inverted
0210
0211      Calculate inverse of A;
0212      Store inverse in A;
0213      Return;
0214      End INVRS;
0215

```

#3 Procedure TREAD

```

0216      Array
0217      LBUF(IDIM/2) stores one row of input picture
0218
0219      Read next record from tape MT;
0220      If EOF encountered
0221      Then write "invalid # of records in picture file";
0222      Stop;
0223      End If;
0224      If # of words read < 32
0225      Then zero-fill rest of input array;
0226      End If;
0227      Repack each byte in LBUF into a word in IBUF, preserving
0228      the order of the bits;
0229      Shift each word left 2 bits to rescale data to 10 bits;
0230      End TREAD;
0231

```

#4 Procedure SOLVE

```

0232      Let g(x) be the cubic spline fit through an equally spaced
0233      line of pixels located at x1, x2, ..., xN and with gray levels
0234

```

```

0239 G ,G ,...,G . Then the 2nd derivatives of g at the pixels
0240 1 2 N
0241 satisfy the following equations:
0242  $g''(x_k) + 4g''(x_{k+1}) + g''(x_{k+2}) = 6 * [(G_{k+2} - 2G_{k+1} + G_k) / (\Delta x)^2]$ 
0243 k k+1 k+2 k+2 k+1 k
0244
0245 for k=1,2,...,N-2. Note:  $\Delta x = \text{delta } x$ .
0246 There are N-2 equations in the N unknowns  $g''(x_k), k=1,2,...,N$ .
0247 k
0248 The additional constraints of 2nd derivative = 0 at the end-
0249 points permit the solution of these equations. In matrix form,
0250 the equations are:
0251  $\begin{pmatrix} 4 & 1 & 0 & \dots & 0 \end{pmatrix} \begin{pmatrix} g''(2) \end{pmatrix}$ 
0252  $\begin{pmatrix} 1 & 4 & 1 & 0 & \dots & 0 \end{pmatrix} \begin{pmatrix} g''(3) \end{pmatrix}$ 
0253  $\begin{pmatrix} 0 & 1 & 4 & 1 & 0 & \dots \end{pmatrix} \begin{pmatrix} g''(4) \end{pmatrix} = Y, Y(k) = 6 * [(G_{k+2} - 2G_{k+1} + G_k) / (\Delta x)^2]$ 
0254  $\begin{pmatrix} \dots \dots \dots 1 \end{pmatrix} \begin{pmatrix} \dots \dots \dots \end{pmatrix}$ 
0255  $\begin{pmatrix} 0 & \dots & 0 & 1 & 4 \end{pmatrix} \begin{pmatrix} g''(N-1) \end{pmatrix}$ 
0256
0257 In this application, N = IDIM and T is the inverse of the
0258 coefficient matrix.
0259 Buffer
0260 I, 0 or # of the row in input array IBUF
0261 containing the desired pixels
0262 J 0 or # of the column in input array IBUF
0263 containing the desired pixels
0264 Array
0265 Y(IDIM-2) temporary storage for expressions on right
0266 hand side of equations for g
0267 If I is non-zero row interpolation
0268 Then compute Y; note  $G_k = \text{IBUF}(I,K)$  and  $\Delta x = \text{XSTEP}$ 
0269 S := T*Y; k
0270 Return;
0271 Else compute Y; column interpolation
0272 note  $G_k = \text{IBUF}(K,J)$  and  $\Delta x = \text{XSTEP}$ 
0273 k
0274 DERIV(J,.) = T*Y;
0275 Return;
0276 End If;
0277 End SOLVE;
0278 #5 Procedure GETCO.
0279 g'' has been determined at the pixel values along either a row
0280 or column of the picture and stored. This determines  $g''(x)$  on
0281 each span (between consecutive input pixels) because each cubic
0282 can be written as a function of  $g''$ :
0283 between pixels at  $x_k$  and  $x_{k+1}$  (span k),
0284
0285 k k+1
0286 Note:  $\Delta x = \text{delta } x$ 
0287  $g''(x_k) = g''(x_{k+1}) / 6 * [(x_{k+1} - x_k)^3 / \Delta x^3 - \Delta x * (x_{k+1} - x_k)] +$ 
0288 k k+1 k+1
0289
0290  $g''(x_{k+1}) / 6 * [(x_k - x_{k+1})^3 / \Delta x^3 - \Delta x * (x_k - x_{k+1})] +$ 
0291 k+1 k k
0292
0293  $G_k * [(x_{k+1} - x_k) / \Delta x] + G_{k+1} * [(x_k - x_{k+1}) / \Delta x]$ 
0294 k k+1 k+1 k
0295 Buffer
0296 I, either 0 or # of row in input array IBUF
0297 containing the desired pixels
0298 J, either 0 or # of column in input array IBUF

```

```

0299           containing the desired pixels
0300           N           span #
0301   If I is non-zero
0302       Then           row interpolation
0303           If N > 1   note g" is 0 at x and x
0304                           1           IDIM
0305               Then A(1) := S(N-1)/6;
0306               Else A(1) := 0;
0307           End If;
0308           If N < IDIM-1
0309               Then A(2) := S(N)/6;
0310               Else A(2) := 0;
0311           End If;
0312           A(3) := IBUF(I,N);
0313           A(4) := IBUF(I,N+1);
0314           Return;
0315
0316   Else           column interpolation
0317       If N > 1   note g" is 0 at x and x
0318                           1           IDIM
0319           Then CCOEF(J,1) := DERIV(J,N-1)/6;
0320           Else CCOEF(J,1) := 0;
0321       End If;
0322       If N < IDIM-1
0323           Then CCOEF(J,2) := DERIV(J,N)/6;
0324           Else CCOEF(J,2) := 0;
0325       End If;
0326       CCOEF(J,3) := IBUF(N,J);
0327       CCOEF(J,4) := IBUF(N+1,J);
0328       Return;
0329   End If;
0330   End GETCO;
0331 #6 Procedure ROW
0332
0333   NPTS points must be inserted between each pair of pixels in
0334   the designated row of input array IBUF. The position values
0335   assigned to these pixels are:
0336
0337   Kth pixel has x coordinate (K-1)*XSTEP.
0338   This allows NPTS pixels to be inserted at subintervals DELTA
0339   in each XSTEP interval.
0340   Buffer
0341       I           # of row in input array IBUF containing
0342                   the desired pixels
0343   If NPTS non-zero   bypass processing if 0 expansion is opted
0344       Then
0345           Call SOLVE;   get 2nd derivatives for cubic spline fit
0346                       across this row
0347           Loop until last pair of pixels processed
0348               Call GETCO;   get the 4 coefficients which specify the
0349                           cubic through current pair of pixels
0350           Loop until NPTS points have been inserted
0351               X := X+DELTA;
0352               Call F;       evaluate cubic at X
0353               Store in output array KBUF;
0354           End Loop;
0355       End Loop;
0356   End If;
0357   Call DSPLAY;       output new row to screen
0358   If IOPT = "N"

```

```

0359         Then Return;
0360         Else Store row on disc LUD;
0361         Return;
0362     End If;
0363     End ROW;
0364 #7 Procedure F
0365     evaluate the cubic interpolation polynomial
0366     Buffer
0367         X,                position of new pixels to be interpolated
0368         I,                span #
0369         DX1,              temporary storage of distance between new
0370                           pixel and left endpoint of span
0371         DX2,              temporary storage of distance between right
0372                           endpoint of span and new pixel
0373         H                 temporary storage of XSTEP
0374     Array
0375         A(4)              weighting coefficients for this span
0376     DX1 := X-(I-1)*H;
0377     DX2 := H-DX1;
0378     F := A(1)*DX2*[DX2**2/H - H] +
0379          A(2)*DX1*[DX1**2/H - H] +
0380          A(3)*DX2/H + A(4)*DX1/H;
0381     Return;
0382     End F;
0383 #8 Procedure DSPLAY
0384
0385     Compute track and sector for next row;
0386     If NPRT = "RA"
0387         Then output JBUF to RAMTEK;
0388         Else output KBUF to CONTAL;
0389     End If;
0390     Return;
0391     End DSPLAY;

```

```

0001
0002 C*****
0003 C*          Written by Ginny Kalb          *
0004 C*          Technology U. S. A.          *
0005 C*          P. O. Box 55333 Ph (301) 292-2592 *
0006 C*          Oxon Hill Station          *
0007 C*          Washington D. C.    20022          *
0008 C*          Rev. 29 Jan 1981          *
0009 C*****
0010 C*          This program is used to explain how to use Program EXPND. *
0011 C*****
0012 C
0013          User's Guide to Cubic Spline Expansion
0014          Program EXPND
0015          This program expands a 64x64 pixel image by row and column cubic spline
0016          interpolation. The expansion factor is selectable, ranging from 1 (no ex-
0017          pansion) to 8 (output image is 505x505 pixels). The output picture is
0018          displayed on either a RAMTEK or COMTAL screen and optionally stored on dis
0019          The user options are specified at run time in response to prompts issued
0020          by the program. The input must be on tape and the tape must be positioned
0021          by the user to the desired picture file prior to program execution.
0022          It is possible to speed up program execution if the user has a data tap
0023          with consecutive images all of which are to be expanded by the same factor
0024          because then the initialization of the algorithm can be skipped.
0025          Note: the input tape is not rewound at the end of the program.
0026          Sample computer-user dialogue:
0027 C  NOTE program defaults to RAMTEK screen on LU 12.
0028          Do you want to change either default? Y/N
0029
0030 U  Y
0031
0032 C  .input CO for COMTAL or RA for RAMTEK
0033
0034 U  RA
0035
0036 C  .input LU #
0037
0038 U  16
0039
0040 C  .specify expansion factor -- e.g. 2 will double input picture which
0041          is 64x64 (or just RETURN and use the default of 4)
0042
0043 U  2
0044
0045 C  .specify input picture tape unit
0046
0047 U  7
0048          delay while initialization is performed
0049 C  .do you want to save output picture? Y/N
0050
0051 U  Y
0052
0053 C  .specify output disc LU and picture number
0054
0055 U  13,2
0056          delay while expansion is performed
0057 C  it is finished
0058          .do you want to expand the next picture? Y/N

```

0059

0060 U Y

0061

0062 C .do you want to save output picture? Y/N

0063

0064

0065

DISTRIBUTION LIST

12 copies	Director Walter Reed Army Institute of Research Walter Reed Army Medical Center ATTN: SGRD-UWZ-C Washington, DC 20307-5100
4 copies	Commander US Army Medical Research and Development Command ATTN: SGRD-RMS Fort Detrick, Frederick, Maryland 21701-5012
12 copies	Defense Technical Information Center (DTIC) ATTN: DTIC-DDAC Cameron Station Alexandria, VA 22304-6145
1 copy	Dean School of Medicine Uniformed Services University of the Health Sciences 4301 Jones Bridge Road Bethesda, MD 20814-4799
1 copy	Commandant Academy of Health Sciences, US Army ATTN: AHS-CDM Fort Sam Houston, TX 78234-6100

END

FILMED

7-85

DTIC